

(19) 日本国特許庁 (J P)

(12) 公表特許公報 (A)

(11) 特許出願公表番号

特表平9-512651

(43) 公表日 平成9年(1997)12月16日

(51) Int.Cl. <sup>6</sup>	識別記号	庁内整理番号	F I
G 0 6 F 9/30	3 1 0	9293-5B	G 0 6 F 9/30
	3 1 0	9462-5B	9/38
			3 1 0 B
			3 1 0 X

審査請求 未請求 予備審査請求 有 (全 32 頁)

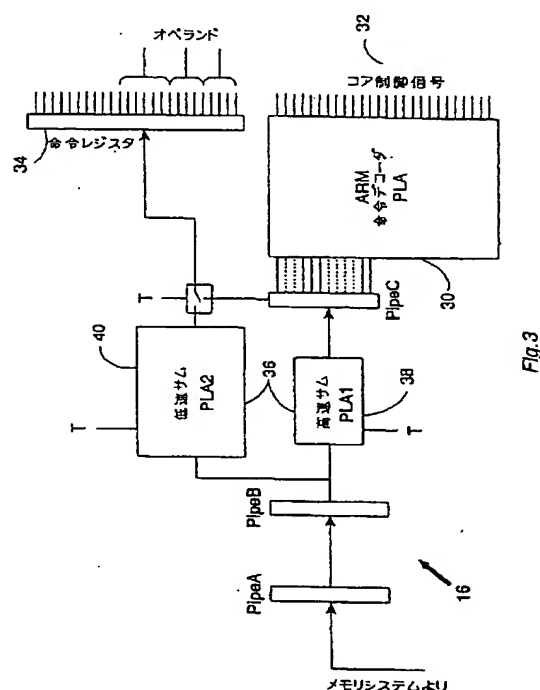
(21) 出願番号 特願平7-528042  
 (86) (22) 出願日 平成7年(1995)2月15日  
 (85) 翻訳文提出日 平成8年(1996)5月30日  
 (86) 国際出願番号 PCT/GB95/00314  
 (87) 国際公開番号 WO95/30187  
 (87) 国際公開日 平成7年(1995)11月9日  
 (31) 優先権主張番号 9408873.9  
 (32) 優先日 1994年5月3日  
 (33) 優先権主張国 イギリス (GB)  
 (81) 指定国 EP(AT, BE, CH, DE, DK, ES, FR, GB, GR, IE, IT, LU, M C, NL, PT, SE), CN, JP, KR, RU

(71) 出願人 アドバンスド リスク マシーンズ リミ  
 テッド  
 イギリス国シービー1 4ジェイエヌ ケ  
 ンブリッジ, チェリー ヒントン, フルバ  
 ーン ロード (番地なし)  
 (72) 発明者 ジャガー, デビッド ビビアン  
 イギリス国シービー1 4ティーエヌ ケ  
 ンブリッジ, チェリー ヒントン, マンド  
 リル クローズ 48  
 (74) 代理人 弁理士 浅村 皓 (外3名)

(54) 【発明の名称】 複数命令セットのマッピング

## (57) 【要約】

複数命令セットを用いるデータ処理システムが説明される。プログラム命令語は、命令パイプライン (6) を経てプロセッサコア (2) へ供給される。第2命令セットのプログラム命令語が該命令パイプラインを通過する時、それらは第1命令セットのプログラム命令語にマップされる。第2命令セットのプログラム命令語のビットサイズは、第1命令セットのそれよりも小さく、第2命令セットは第1命令セットのサブセットである。小さいビットサイズはコード密度を改善し、一方第2命令セットの、第1命令セットのサブセットとしての性質は、1対1マッピングが効率的に行われうようにするので、第2命令セットのための専用命令デコーダの必要を回避する。



## 【特許請求の範囲】

1. データ処理装置であって、該装置が、  
複数のコア制御信号に応答するプロセッサコアと、  
第1命令セットのXビットプログラム命令語のPビットをデコードして前記コア制御信号を発生するデコード手段と、  
該デコード手段への命令プログラム語が通過する命令パイプラインと、  
該命令パイプラインを通過する第2命令セットのYビットプログラム命令語に応答して、該Yビットプログラム命令語のQビットを、前記デコード手段によるデコード用の対応するXビットプログラム命令語の前記Pビットにマップする第1マッピング手段と、を含み、  
YがXより小さく、かつ前記第2命令セットが前記第1命令セットのサブセットである、  
データ処理装置。
2. 前記プロセッサコアにより実行されるXビット命令を保持するための命令レジスタを含み、該プロセッサコアが、該命令レジスタからオペランド値を読み取る、請求項第1項記載の装置。
3. 前記命令パイプライン内の前記Yビットプログラム命令語内のオペランド値を、前記対応するXビットプログラム命令語内の対応する位置にマップし、かつ該マップされたオペランド値を前記プロセッサコアによる使用のために前記命令レジスタ内に記憶させる、第2マッピング手段を含む、請求項第2項記載の装置。
4. 前記第2マッピング手段が、前記Yビットプログラム命令語を、前記対応するXビットプログラム命令語の完全なバージョンにマップし、かつ該対応するXビットプログラム命令語の該完全なバージョンを前記命令レジスタ内に記憶させる、請求項第3項記載の装置。
5. 命令プログラム語が前記命令パイプラインを複数の処理サイクルにわたって通過し、該処理サイクルの1つがデコードサイクルであり、前記デコード手段が該デコードサイクルの終了までに前記コア制御信号を発生する演算を行い、前

記第1マッピング手段が該デコード部分の第1部分中に前記対応するXビットプログラム命令語の前記Pビットを発生する演算を行い、前記デコード手段が前記デコードサイクルの前記終了までになお前記コア制御信号を発生することを可能ならしめる、請求項第1項から第4項までのいずれかに記載の装置。

6. 前記第2マッピング手段が、前記デコードサイクルの終了までに前記命令レジスタ内に前記マップされたオペランド値を記憶させる演算を行う、請求項第3項および第5項記載の装置。

7. 前記第1マッピング手段が前記第2マッピング手段と並列に演算を行う、請求項第3項記載の装置。

8. 前記プロセッサコアが、前記第1命令セットによって用いられ且つある前記Xビットプログラム命令語内においてレジスタオペランドとして定義される複数のレジスタを有し、また前記第2命令セットが、ある前記Yビットプログラム命令語内においてレジスタオペランドとして定義される前記レジスタのサブセットを用いる、請求項第1項から第7項までのいずれかに記載の装置。

9. 前記第2マッピング手段が、前記Yビットプログラム命令語の前記レジスタオペランドを拡張し、前記Xビットプログラム命令語の前記レジスタオペランドを生ぜしめる、請求項第3項および第8項記載の装置。

10. 前記Xビットプログラム命令語の前記オペランドが、前記Yビットプログラム命令語の前記オペランドより大きい範囲を有し、前記第2マッピング手段の上位のゼロが、前記Yビットプログラム命令語からの前記オペランドを拡張して前記Xビットプログラム命令語の前記オペランドを生じる、請求項第3項記載の装置。

11. PがXより小さい、請求項第1項から第10項までのいずれかに記載の装置。

12. QがPよりも小さいか、またはPに等しい、請求項第1項から第11項までのいずれかに記載の装置。

13. Xが32であり、Yが16である、請求項第1項から第12項までのいずれかに記載の装置。

14. 前記装置が集積回路である、請求項第1項から第13項までのいずれか

に記載の装置。

15. データ処理方法であって、該方法が、

複数のコア制御信号に応答するプロセッサコアと、

デコード手段により第1命令セットのXビットプログラム命令語のPビットをデコードして、プロセッサコアを制御するコア制御信号を発生させるステップと

、

命令プログラム語を命令パイプラインを経て前記デコード手段へ送るステップと、

前記命令パイプラインを通過する第2命令セットのYビットプログラム命令語のQビットを、前記デコード手段によるデコードのために、対応するXビットプログラム命令語の前記Pビットにマップするステップと、を含み、

YがXより小さく、かつ前記第2命令セットが前記第1命令セットのサブセットである、

データ処理方法。

## 【 発 明 の 詳 細 な 説 明 】

複数命令セットのマッピング

本発明は、データ処理の分野に関する。特に、本発明は、プログラム命令語の複数セットを用いるデータ処理に関する。

データ処理システムは、プログラム命令語の制御を受けて演算を行うプロセッサコアによって演算を行い、プログラム命令語は、デコードされた時、該プログラム命令語によって指定された処理を遂行するために必要な演算を行うプロセッサによりさまざまな素子を制御するための、コア制御信号を発生させる働きを有する。

1つより多くの命令セットによって演算するシステムの使用は公知である（例えば、ディジタル・イクイプメント・コーポレーション（Digital Equipment Corporation）のVAX11コンピュータは、VAX命令モードと、該コンピュータによる早期のPDP11コンピュータ用命令のデコードを可能にする互換性モードと、を有する）。そのようなシステムには通常、それぞれの命令セット用の別個の命令デコーダが組み込まれてきた。命令デコーダは、比較的複雑かつ大形の回路素子である。

集積回路が占有するスペースは高価である。集積回路は小さいほど、安価に製造され、製造歩留りが高くなる。さらに、集積回路上においては、他の回路素子による使用のために、追加のスペースが利用されうるようにもされる。集積回路の寸法を縮減する手段は大きい利点を有する。

1つの特徴から見た時、本発明はデータ処理装置を提供し該装置は、

複数のコア制御信号に応答するプロセッサコアと、

第1命令セットのXビットプログラム命令語のPビットをデコードして前記コア制御信号を発生するデコード手段と、

該デコード手段への命令プログラム語が通過する命令パイプラインと、

該命令パイプラインを通過する第2命令セットのYビットプログラム命令語に  
応答して、該Yビットプログラム命令語のQビットを、前記デコード手段による

デコード用の対応するXビットプログラム命令語の前記Pビットにマップする第

1 マッピング手段と、を含み、

YはXより小さく、かつ前記第2命令セットは前記第1命令セットのサブセットである。

本発明は、命令パイプラインが組み込まれたシステムに適用され、命令を、第2命令セットから第1命令セットへ、それらが該命令パイプラインを通過する時にマップするのに役立つ。第2命令セットのプログラム命令語を、第1命令セットのそれらへ翻訳すれば、第2命令デコーダの必要が回避され、より簡単でより効率的な、プロセッサコアの残部の実施例が可能となる。本発明においては、もし第2命令セットが第1命令セットのサブセットであるようにされれば、1対1マッピングが可能となり、これは、システムパフォーマンスを制限しないように、十分に規則的かつ高速に実現されうることが認識される。さらに、本発明においては、前記デコード手段を実際に駆動するためには、第2命令セットの命令のあるビットのみを、第1命令セット内の命令語の対応するビットにマップする必要があることが認識される。これは、マッピングが行われる速度を増大せしめる。

命令デコーディングの速度は、実施例においては、前記プロセッサコアにより実行されるXビット命令を保持するための命令レジスタの具備によって増大せしめられ、該プロセッサコアは、該命令レジスタからオペランド値を読取る。

このようにして、命令デコーダによるデコーディングを必要とせず、プロセッサコアの作用のみを受ける命令内からのオペランドは、命令デコーダを通過するを要せず、命令から直接読取られうる。

この構成を容易ならしめ、それによって、デコード手段を駆動するのに必要な臨界Pビットを高速で発生させ、しかも完全なXビットプログラム命令語を、プロセッサコアによって修正されずに用いられるように後に発生させるために、本発明の実施例は、前記命令パイプライン内の前記Yビットプログラム命令語内のオペランド値を、前記対応するXビットプログラム命令語内の対応する位置にマップし、かつ該マップされたオペランド値を前記プロセッサコアによる使用のために前記命令レジスタ内に記憶させる、第2マッピング手段を含む。

該第2マッピング手段は、前記プロセッサコアが必要とするオペランド値のマ

ッピングの働きのみをなすものでありうる。しかし、前記命令デコーダの設計は、前記プロセッサコアの設計から効果的に分離されうるので、前記第2マッピング手段が、前記Yビットプログラム命令語を、前記対応するXビットプログラム命令語の完全なバージョンにマップし、かつ該対応するXビットプログラム命令語の該完全なバージョンを前記命令レジスタ内に記憶させるようにすることによって、いずれの前記設計の将来の独立した変更も容易となる。

プログラム命令語が命令パイプラインを通過する時、通常、デコード機構にあるタイミングリミットが課せられる。現存のハードウェアとの統合をより容易にするために、実施例内においては、命令プログラム語が前記命令パイプラインを複数の処理サイクルにわたって通過し、該処理サイクルの1つがデコードサイクルであり、前記デコード手段が該デコードサイクルの終了までに前記コア制御信号を発生する演算を行い、前記第1マッピング手段が該デコード部分の第1部分中に前記対応するXビットプログラム命令語の前記Pビットを発生する演算を行い、前記デコード手段が前記デコードサイクルの前記終了までになお前記コア制御信号を発生することを可能ならしめる。

前記デコード手段を駆動するために、Yビットプログラム命令語のあるビットのみがマップされる必要があるという特徴は、そのようなタイミング要求が満たされることを可能ならしめる。

あまり厳しくない関連する要求は、オペランド値がいつ得られなければならないかの要求である。本発明の実施例は、前記第2マッピング手段が、前記デコードサイクルの終了までに前記命令レジスタ内に、前記マップされたオペランド値を記憶させる演算を行うようにすることによって、このゆるやかな要求を用いる。

前記第1マッピング手段と前記第2マッピング手段との演算は、独立ならしめられうることを認識すべきである。このシステムの全体的パフォーマンスは従って、もしそれらが並列に演算を行えば改善される。

前記第2命令セットのプログラム命令語のサイズが小さくなると、備えられるべき機能の修正が必要になる。好ましくは、本システムは、前記プロセッサコアが、前記第1命令セットによって用いられ且つある前記Xビットプログラム命令

語内においてレジスタオペランドとして定義される複数のレジスタを有し、また前記第2命令セットが、ある前記Yビットプログラム命令語内においてレジスタオペランドとして定義される前記レジスタのサブセットを用いるように構成される。

前記第1命令セットによって用いられる前記レジスタのサブセットの、前記第2命令セットによる使用は、命令間の1対1マッピングがなお行われることを可能ならしめ、しかも前記第2命令セットのビットサイズを小さくする。

命令セット間でのレジスタの異なる様式のハンドリングを処理するために、前記第2マッピング手段は、前記Yビットプログラム命令語の前記レジスタオペランドを拡張し、前記Xビットプログラム命令語の前記レジスタオペランドを生ぜしめる。

同様にして、前記第2命令セット内の他のオペランドは、前記第1命令セットのオペランドと比較して減少した範囲を有し、それらの最左端は、前記第2マッピング手段によるマッピング中に、ゼロを拡張される。

もう1つの特徴から見た時、本発明はデータ処理方法を提供し該方法は、複数のコア制御信号に応答するプロセッサコアと、

デコード手段により第1命令セットのXビットプログラム命令語のPビットをデコードして、プロセッサコアを制御するコア制御信号を発生させるステップと、

命令プログラム語を命令パイプラインを経て前記デコード手段へ送るステップと、

前記命令パイプラインを通過する第2命令セットのYビットプログラム命令語のQビットを、前記デコード手段によるデコードのために、対応するXビットプログラム命令語の前記Pビットにマップするステップと、を含み、

YはXより小さく、かつ前記第2命令セットは前記第1命令セットのサブセットである。

次に、添付図面を参照しつつ、本発明の実施例を単なる例として説明する。添付図面において、



第 1 図は、プロセッサコアおよびメモリシステムを組み込んだデータ処理装置を概略的に示し、

第 2 図は、単一命令セットを有するシステム用の命令および命令デコーダを概略的に示し、

第 3 図は、2つの命令セットを有するシステムに用いられる命令パイプラインおよび命令デコーダを示し、

第 4 図は、Xビットプログラム命令語のデコーディングを示し、

第 5 図および第 6 図は、Yビットプログラム命令語の、Xビットプログラム命令語へのマッピングを示し、

第 7 図は、Xビット命令セットを示し、

第 8 図は、Yビット命令セットを示し、

第 9 図は、第 1 命令セットおよび第 2 命令セットにとって利用可能な処理レジスタを示す。

第 1 図は、Yビットメモリシステム 4 に結合せしめられたプロセッサコア 2 を含む、(集積回路の一部として形成された) データ処理システムを示す。この場合、Y は 16 に等しい。

プロセッサコア 2 は、レジスタバンク 6 と、ブースの乗算器 8 と、バレルシフタ 10 と、32ビット論理演算装置 12 と、書込みデータレジスタ 14 と、を含む。プロセッサコア 2 とメモリシステム 4 との間には、命令パイプライン 16 と、命令デコーダ 18 と、読取りデータレジスタ 20 と、が配置されている。プロセッサコア 2 の一部であるプログラムカウンタレジスタ 22 は、メモリシステム 4 をアドレス指定するように示されている。プログラムカウンタインクリメンタ 24 は、それぞれの命令が実行され、かつ新命令が命令パイプライン 16 のためにフェッチされなければならない時、プログラムカウンタレジスタ 22 内のプログラムカウンタ値を増加させる働きをする。

プロセッサコア 2 は、さまざまな機能ユニット間に N ビットデータ経路 (この場合は 32 ビットデータ経路) を取り入れる。演算に際しては、命令パイプライン 16 内の命令が命令デコーダ 18 によってデコードされ、該デコーダはさまざま

まなコア制御信号を発生し、該コア制御信号はプロセッサコア2内のさまざまな機能素子へ送られる。これらのコア制御信号に応答して、前記プロセッサコアのさまざまな部分は、32ビット乗算、32ビット加算、および32ビット論理演算のような、32ビット処理演算を行う。

レジスタバンク6は、現プログラミング状態レジスタ26と、セーブプログラミング状態レジスタ28と、を含む。現プログラミング状態レジスタ26は、プロセッサコア2のためのさまざまな条件および状態フラグを保持する。これらのフラグには、処理モードフラグ（例えば、システムモード、ユーザモード、メモリアポートモード、など）と、算術演算におけるゼロ結果、桁上げ、などの発生を表示するフラグと、が含まれる。セーブプログラミング状態レジスタ28（これはバンクをなす複数のこのようなセーブプログラミング状態レジスタの1つでありうる）は、もし処理モードスイッチをトリガする異常が起こったならば、現プログラミング状態レジスタ26の内容を一時的に記憶するために用いられる。このようにして、異常処理は、より高速かつより効率的に行われうる。

現プログラミング状態レジスタ26内には、命令セットフラグTが含まれる。この命令セットフラグは、命令デコーダ18およびプログラムカウンタインクリメンタ24へ供給される。この命令セットフラグTがセットされた時、システムは、第2命令セットの命令（すなわち、この場合は16ビットプログラム命令語であるYビットプログラム命令語）によって演算する。命令セットフラグTは、プログラムカウンタインクリメンタ24を制御して、第2命令セットにより演算せしめられた時、小さいインクリメントステップを採用せしめる。これは、第2命令セットのプログラム命令語が小さく、従ってメモリシステム4の記憶場所内において狭い間隔を有することに符合している。

前述のように、メモリシステム4は、16ビットデータバスを経て読取りデータレジスタ20および命令パイプライン16に接続された、16ビットメモリシステムである。そのような16ビットメモリシステムは、高性能の32ビットメモリシステムに比し簡単かつ経済的である。そのような16ビットメモリシステムを用いると、単一サイクル内に16ビットプログラム命令語がフェッチされう

る。しかし、もし（命令セットフラグ T によって表示される）第 2 命令セットからの 32 ビット命令が用いられるならば、命令パイプライン 16 に対して単一の 32 ビット命令を回復するために 2 回の命令フェッチが必要になる。

必要なプログラム命令語がメモリシステム 4 から回復された時、それらは命令デコーダ 18 によってデコードされ、命令が 16 ビット命令であるか、32 ビッ

ト命令であるかにかかわらず、プロセッサコア 2 内において 32 ビット処理が開始される。

第 1 図においては、命令デコーダ 18 は単一ブロックとして示されている。しかし、1 つより多くの命令セットを処理するために、命令デコーダ 18 は、第 2 図および第 3 図に関連して説明されるように、もっと複雑な構造を有する。

第 2 図は、単一命令セットに対処するための命令パイプライン 16 および命令デコーダ 18 を示す。この場合、命令デコーダ 18 は、32 ビット命令をデコードする演算を行う第 1 デコード手段 30 のみを含む。このデコード手段 30 は、プログラム可能論理アレイ（PLA）を用いて第 1 命令セット（ARM 命令セット）をデコードし、プロセッサコア 2 へ供給される複数のコア制御信号を発生する。現在デコードされている（すなわち、現在のコア制御信号 32 を生じる）プログラム命令語もまた、命令レジスタ 34 内に保持されている。プロセッサコア 2 内の機能素子（例えば、ブースの乗算器 8 またはレジスタバンク 6）は、該素子の処理演算のために必要なオペランドを、直接この命令レジスタ 34 から読取る。

このような構造の演算の特徴は、第 1 デコード手段 30 が、該第 1 デコード手段が演算するクロックサイクルの初期に、ある入力（パイプライン段 P i p e C から出る実線として示されている P ビット）を必要とすることである。これは、コア制御信号 32 が、プロセッサコア 2 内の必要な素子を駆動するのにちょうどよい時に発生せしめられることを保証するためである。第 1 デコード手段 30 は、比較的大形かつ低速のプログラム可能論理アレイ構造であるので、そのようなタイミングの考慮が重要である。

命令デコーディングを行うためのそのようなプログラム可能論理アレイ構造の

設計は、本技術分野における通常のものである。入力組は、それらの入力から発生せしめられるべき所望の出力と共に定められる。その際、市販のソフトウェアが、指定された入力組から指定された出力組を発生するPLA構造を考案するのに用いられる。

第3図は、第1命令セットおよび第2命令セットのデコーディングを扱うように改変された第2図のシステムを示す。命令セットフラグTによって第1命令セ

ットが選択された時は、このシステムは、第2図に関連して説明されたように演算する。命令セットフラグTが、命令パイプライン16内の命令が第2命令セットからのものであることを表示している時は、第2デコード手段36が活動状態になる。

この第2デコード手段は、高速PLA38および並列の低速PLA40を用いて16ビット命令（サム（Thumb）命令）をデコードする。高速PLA38は、16ビットサム命令の諸ビットのサブセット（Qビット）を、第1デコード手段30を駆動するのに必要な対応する32ビットARM命令のPビットにマップする働きをする。このマッピングを受けるためには比較的少数のビットが必要とされるので、高速PLA38は比較的浅くてもよい。第1デコード手段がPipeCの内容に応答してコア制御信号32を発生するための十分な時間を与えうるほど高速で演算を行う。高速PLA38は、完全な命令をマッピングする不必要な時間を費やすことなく、第1デコード手段のための対応する32ビット命令の臨界ビットを「捏造する」ために働くものと考えられうる。

しかし、完全な32ビット命令は、もしプロセッサコア2が根本的な変更および顕著な追加の回路素子なしに演算しうるとすれば、プロセッサコア2によってなお必要とされる。時間的に臨界的なマッピングが高速PLA38により処理されたとすると、並列に接続された低速PLA40は、16ビット命令を対応する32ビット命令にマップして、これを命令レジスタ34内に配置する働きをする。このさらに複雑なマッピングは、高速PLA38および第1デコード手段30が演算するのに要する全時間にわたって行われうる。重要な因子は、32ビット命令が、プロセッサコア2に作用するコア制御信号32に応答して命令レジスタ

34から読取られるべきオペランドに対して十分な時間命令レジスタ34内に存在すべきことである。

第2命令セットをデコードする時の第3図のシステムの全体的な演算は、第2命令セットからの16ビット命令を、それらが命令パイプライン16に沿って進んでいる時に、第1命令セットからの32ビット命令に翻訳することである。これは、第2命令セットからの命令の、第1命令セット内の命令への1対1マッピングの存在を保証するために、第2命令セットを第1命令セットのサブセットと

することにより、実際に可能ならしめられる。

命令セットフラグTの具備は、第2命令セットが第1命令セットに対して非直交的であることを可能にする。これは、第1命令セットが、直交するさらなる命令セットが検出され且つデコードされることを可能にするために用いられうる自由ビットのない、現存の命令である状況においては特に有用である。

第4図は、32ビット命令のデコーディングを示す。第4図の上部には、逐次処理クロックサイクルが示されており、それらのサイクルにおいては、フェッチ演算、デコード演算、および最後に実行演算が行われる。もし特定の命令（例えば、乗算命令）が必要とするならば、1つまたはそれ以上の追加の実行サイクルが加えられる。

32ビット命令42は、複数の異なるフィールドから構成される。これらのフィールドの間の境界は、後に第7図に示されるように、異なる命令に対しては異なる。

命令42内のある諸ビットは、1次デコード相内におけるデコーディングを必要とする。これらのPビットは、ビット4ないし7、20および22ないし27である。これらは、第1デコード手段30により必要とされる、また高速PLA38により「捏造」されなければならない、ビットである。これらのビットは、第1デコード手段へ印加され、それによってデコードされて、デコードサイクルの第1部分の終了までに適切なコア制御信号32を発生しなければならない。もし必要ならば、完全な命令のデコーディングは、デコードサイクルの終了まで長くかかってもよい。デコードサイクルの終了時には前記実行サイクル中に、命令

レジスタ34から命令内のオペランドが、プロセッサ2によって読取られる。これらのオペランドは、レジスタ指定子、オフセット、または他の変数でありうる。

第5図は、16ビット命令の例の、32ビット命令へのマッピングを示す。太い線は、32ビット命令内のPビットへのマッピングを必要とする、16ビット命令内のQビットから発しており、従ってそれらは第1デコード手段30へ印加されうる。これらのビットの大部分は、直接コピーされるか、または簡単なマッピングを受けることがわかる。16ビット命令内のオペランドR<sub>n'</sub>、R<sub>d</sub>およびイミディエイト(immediate)は、それらの最上位端をゼロにより

パッドして、32ビット命令を満たす必要がある。このパディングは、32ビット命令のオペランドが、16ビット命令のオペランドより大きい範囲を有するので必要とされる。

第5図の下部に与えられている32ビット命令の一般化された形式からわかるように、該32ビット命令は、16ビット命令によって表されるその命令のサブセットよりもかなり大きい柔軟性を許容する。例えば、該32ビット命令は、該命令を条件付きで実行可能とする条件コードC<sub>ond</sub>の後にある。これとは対照的に、前記16ビット命令は自身の中に条件コードを有せず、それらがマップされる対象である前記32ビット命令の条件コードは、条件付き実行状態「常に」と等価である「1110」の値にセットされる。

第6図は、もう1つのそのような命令マッピングを示す。この場合の16ビット命令は、第5図に示されているものとは異なるタイプのロード／記憶命令である。しかし、この命令はなお、前記32ビット命令セットの単一データ転送命令のサブセットである。

第7図は、前記32ビット命令セットのための11の異なるタイプのフォーマットを概略的に示す。これらの命令を以下に順次列記する。

1. データ処理PSR転送。
2. 乗算。
3. 単一データスワップ。

4. 単一データ転送。
5. 不定。
6. ブロックデータ転送。
7. 分岐。
8. コプロセッサデータ転送。
9. コプロセッサデータ演算。
10. コプロセッサレジスタ転送。
11. ソフトウェア割込み。

この命令セットの十分な説明は、アドバンスト・RISC・マシNZ・リミテツド (Advanced RISC Machines Limited) により

生産されているARM6プロセッサのデータシートに見出されうる。第7図内において強調されている命令は、第5図および第6図に示されている命令である。

第8図は、前記32ビット命令セットに加えて備えられる前記16ビット命令セットを示す。この命令セット内において強調されている命令は、第5図および第6図のそれぞれに示されている命令である。この16ビット命令セット内の命令は、それら全てが単一32ビット命令にマップされ、それによって32ビット命令セットのサブセットを形成しうるように選択されている。

この命令セット内のそれぞれの命令を順次考察すると、諸フォーマットは以下のような指定を行う。

フォーマット1: Op = 0, 1。両Opは、条件コードフラグをセットする。

0: ADD Rd, Rs, #Immediate3

1: SUB Rd, Rs, #Immediate3

フォーマット2: Op = 0, 1。両Opは、条件コードフラグをセットする。

0: ADD Rd, Rs, Rn

1: SUB Rd, Rs, Rn

フォーマット3: 3つの演算コード。大きいイミディエイトを作るのに用いられる。

1 = ADD Rd, Rd, #Immediate8 << 8

2 = ADD R d , R d , # I m m e d i a t e 8 < < 1 6

3 = ADD R d , R d , # I m m e d i a t e 8 < < 2 4

フォーマット 4 : O p は 3 つの演算コードを与え、全ての演算は MOV S R d , R s S H I F T # I m m e d i a t e 5 であり、S H I F T は、

0 が L S L であり

1 が L S R であり

2 が A S R であり

A R M に関し定められているようなゼロによるシフト。

フォーマット 5 : O p 1 \* 8 + O p 2 は 3 2 の A L U 演算コードを与え、R d = R d o p R n 。全ての演算は条件コードフラグをセッ

トする。

演算は以下の通りである

AND, OR, EOR, BIC (AND NOT), NEG  
ATE, CMP, CMN, MUL, TST, TEQ, MOV  
, MVN (NOT), LSL, LSR, ASR, ROR  
欠けているのは ADC, SBC, MULL

A R M に関し定められているようなゼロによる且つ 3 1 より  
大きいシフト

8 つの特殊な演算コード、LO は R e g 0 - 7 を指定し、H  
I はレジスタ 8 - 1 5 を指定する

S P E C I A L は C P S R または S P S R である

MOV H I , L O (隠れレジスタから可視レジスタへ  
移動)

MOV L O , H I (可視レジスタから隠れレジスタへ  
移動)

MOV H I , H I (例えば手続き戻り)

MOV S H I , H I (例えば例外戻り)



```
MOV S    H I, L O (例えば割込み戻り、SUB S, H
            I, H I, # 4 でありうる)
```

MOV SPECIAL, LO(MSR)

MOV LO, SPECIAL (MRS)

CMP HI, HI (スタックリミットチェック)

## 8つの自由演算コード

フォーマット 6 : Op は 4 つの演算コードを与える。全ての演算は条件コードフラグをセットする。

```
0 : MOV    Rd, # Immediate 8
```

```
1 : CMP    Rs, #Immediate8
```

```
2 : ADD    Rd, Rd, # Immediate 8
```

A D D は、A D D R d , R s , # I m m e d i a t e 5 と  
交換可能である。

フォーマット 7 : 語 P C + O f f s e t ( 2 5 6 語、1 0 2 4 バイト) をロードする。オフセットはアラインされた語でなければならないことに注意すべきである。

L D R    R d ,    [ P C ,    # + 1 0 2 4 ]

この命令は、次のリテラルプールにアクセスするため、定数、アドレスなどをロードするために用いられる。

フォーマット 8 :    S P ( r 7 ) + 2 5 6 語 ( 1 0 2 4 バイト ) から の 語 を ロードし記憶せしめる

SP ( r 7 ) + 2 5 6 バイトからのバイトをロードし記憶せしめる

L D R    R d ,    [ S P ,    # + 1 0 2 4 ]

L D R B    R d ,    [ S P ,   # + 2 5 6 ]

これらの命令は、スタックおよびフレームアクセスのためのものである。

フォーマット 9 : 語 (またはバイト)、符号付き 3 ビットイミディエイトオ

フセット（増／減後）、強制ライトバック、をロードし記憶せしめる

Lはロード／記憶であり、Uはアップ／ダウン（オフセットの加算／減算）であり、Bはバイト／語である

L D R { B } R d , [ R b ] , # + / - O f f s e t 3

S T R { B } R d , [ R b ] , # + / - O f f s e t 3

これらの命令はアレイアクセス用として意図されている

オフセットはバイトに対しては0-7を、語に対しては4-28をエンコードする

フォーマット10：符号付きレジスタオフセット（増／減前）を有する語（またはバイト）、ライトバックなし、をロードし記憶せしめる

Lはロード／記憶であり、Uはアップ／ダウン（オフセットの加算／減算）であり、Bはバイト／語である

L D R R d , [ R b , + / - R o , L S L # 2 ]

S T R R d , [ R b , + / - R o , L S L # 2 ]

L D R B R d , [ R b , + / - R o ]

S T R B R d , [ R b , + / - R o ]

これらの命令は、ベース+オフセットポインタアクセス用として意図され、8ビットのMOV、ADD、SUBと組み合わせられるとかなり高速のイミディエイトオフセットアクセスを与える。

フォーマット11：符号付き5ビットイミディエイトオフセット（増／減前）を有する語（またはバイト）、ライトバックなし、をロードし記憶せしめる

Lはロード／記憶であり、Bはバイト／語である

L D R { B } R d , [ R b , # + O f f s e t 5 ]

S T R { B } R d , [ R b , # + O f f s e t 5 ]

これらの命令は、構造アクセス用として意図されている

オフセットはバイトに対しては0-31を、語に対しては0、4-124をエンコードする

フォーマット12：倍数（強制ライトバック）をロードし記憶せしめる

LDMIA Rb!, {Rlist}

STMIA Rb!, {Rlist}

Rlistはレジスタr0-r7を指定する

これらの命令のサブクラスは、1対のサブルーチンコールおよびリターン命令である。

LDMに対し、もしr7がベースであり、ビット7が

Rlist内にセットされれば、PCがロードされる

STMに対し、もしr7がベースであり、ビット7が

Rlist内にセットされれば、LRが記憶される

もしr7がベースレジスタとして用いられれば、spが代わりに用いられる

双方の場合に完全降順スタックが実現され、すなわちLDMはARMのLDMFDと同様になり、STMはARMのSTMFDと同様になる

それゆえブロックコピーに対し、r7をエンドポインタとして用いる

もしr7がベースでなければ、LDMおよびSTMは、ARMのLDMIA、STMIAと同様になる

フォーマット13：アドレスをロードする。この命令は、8ビット符号なし定数をPCまたはスタックポインタに加算し、結果を宛先レジスタ内に記憶させる。

ADD Rd, sp, +256bytes

ADD Rd, pc, +256words (1024bytes)

SPビットは、SPまたはPCがソースであるか否かを表示

する

もしSPがソースであり、かつr7が宛先レジスタとして指定されれば、SPは宛先レジスタとして用いられる。

フォーマット14：条件付き分岐、+/-128バイト、この場合condは（ARMに関してと同様の）条件コードを定め、cond=15はSWIとしてエンコードする（256のみで多分十分である）。

フォーマット15：長い分岐およびリンクのビット22:12をセットする。

MOV lr, #offset<<12。

フォーマット16：長い分岐およびリンクを行う。演算は、SUB newlr, pc, #4; ORR pc, oldlr, #offset<<1である。newlrおよびoldlrは、この演算の前および後のlrレジスタを意味する。

前述のように、16ビット命令セットは32ビット命令セットに比し減少したオペランド範囲を有する。これに符合して、16ビット命令セットは、完全な32ビット命令セット用に具備されたレジスタ6（第1図参照）のサブセットを用いる。第9図は、16ビット命令セットにより用いられるレジスタの該サブセットを示す。

【 図 1 】

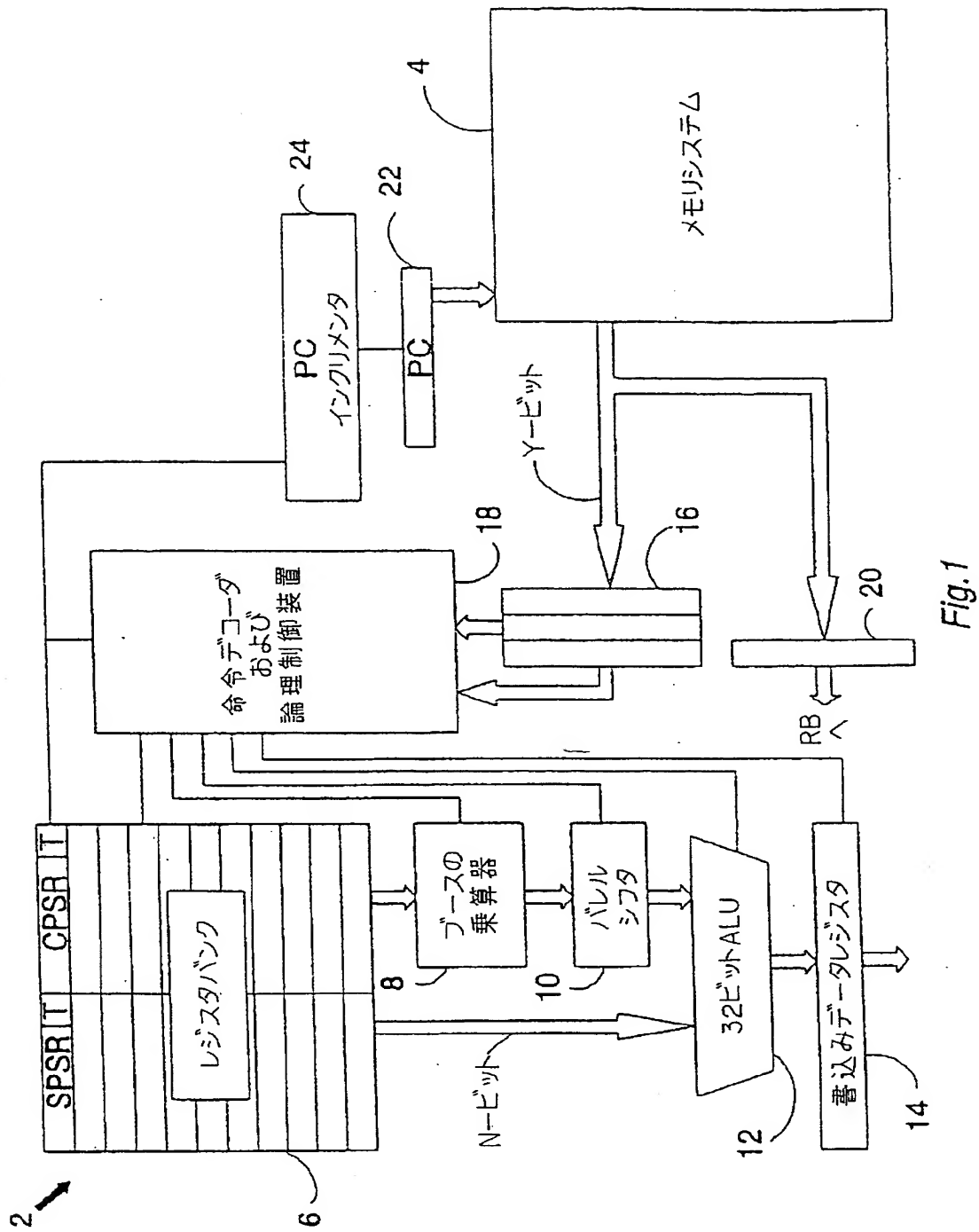


Fig.1

【 図 2 】

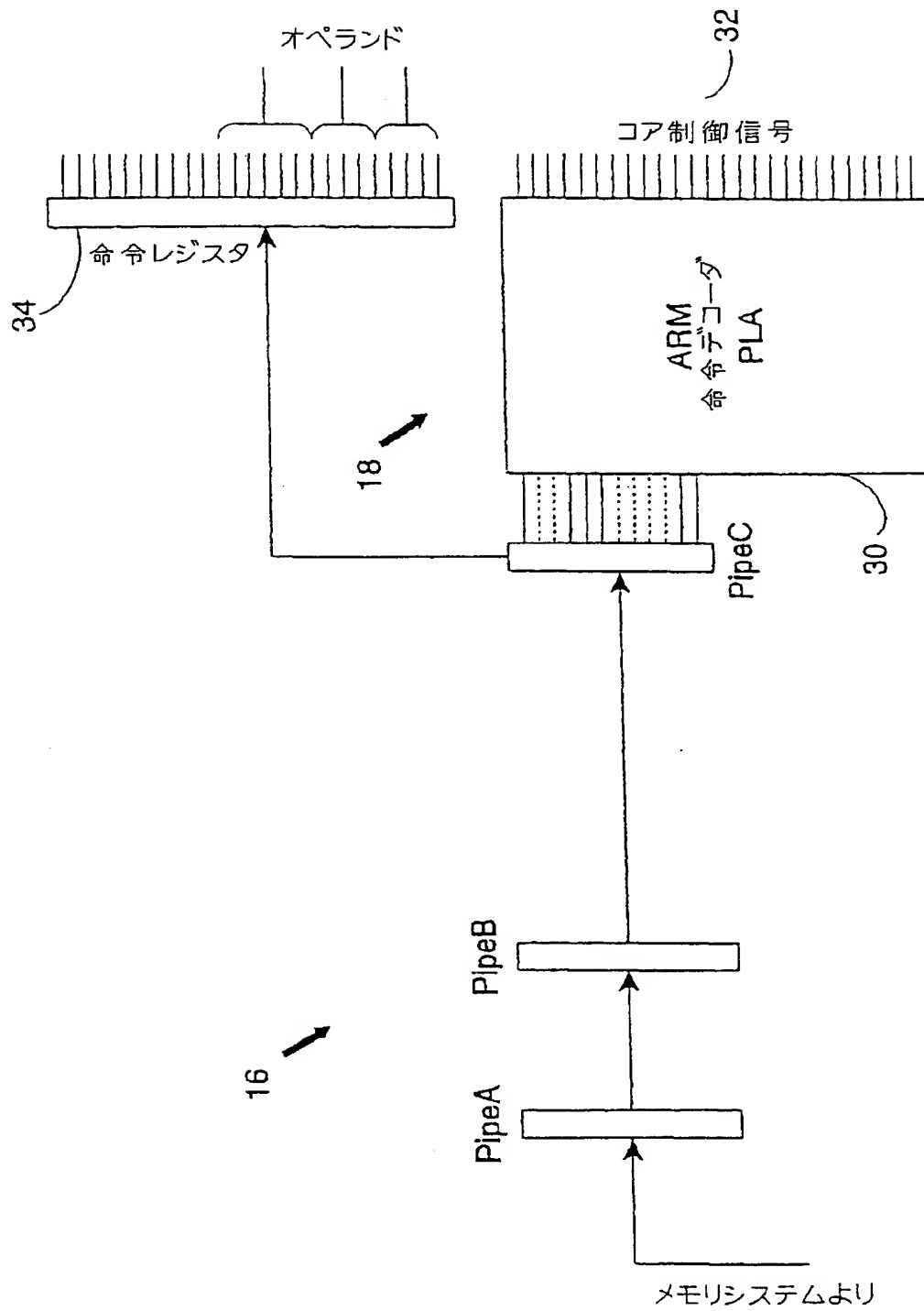


Fig.2

【 図 3 】

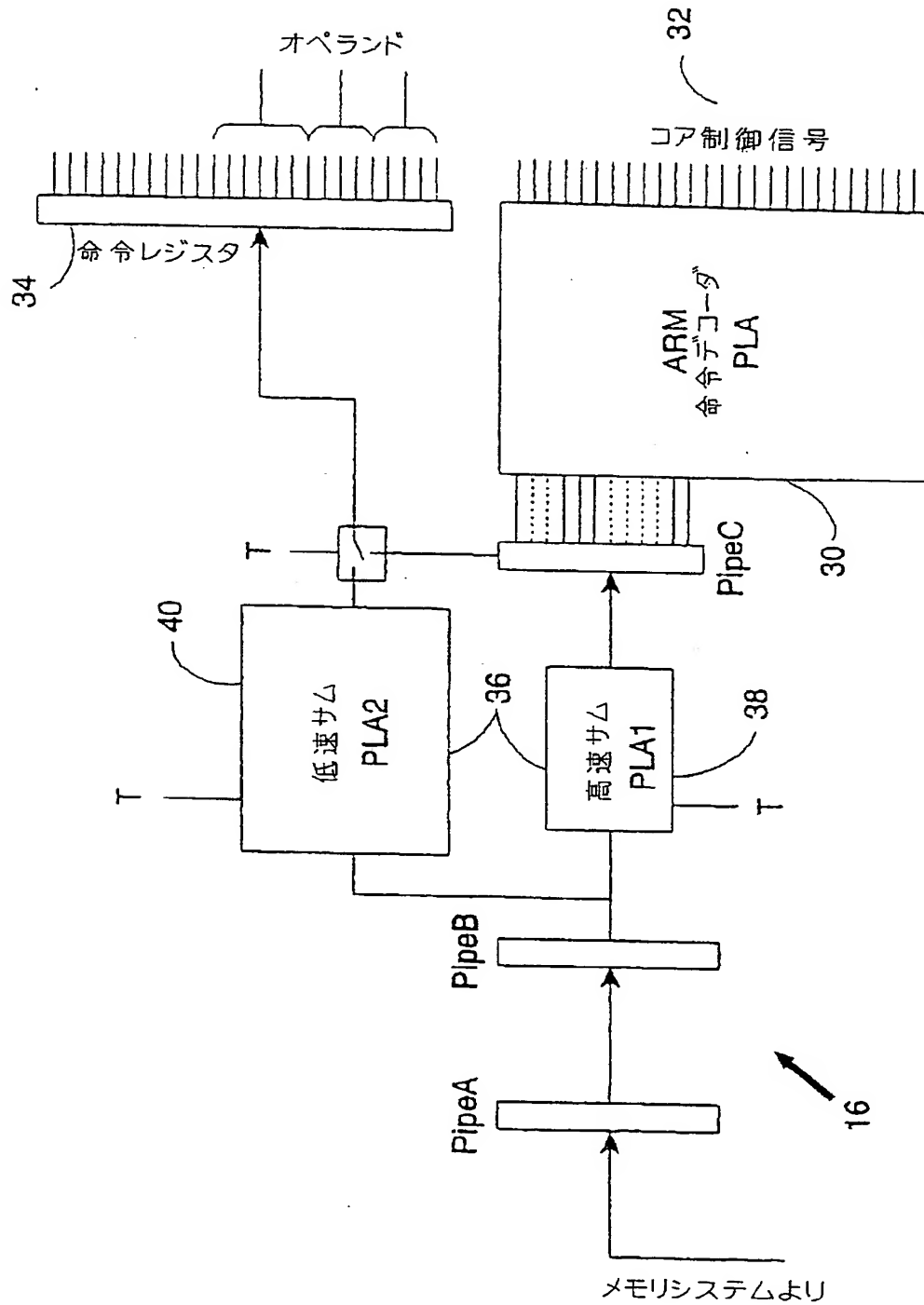


Fig.3

【 図 4 】

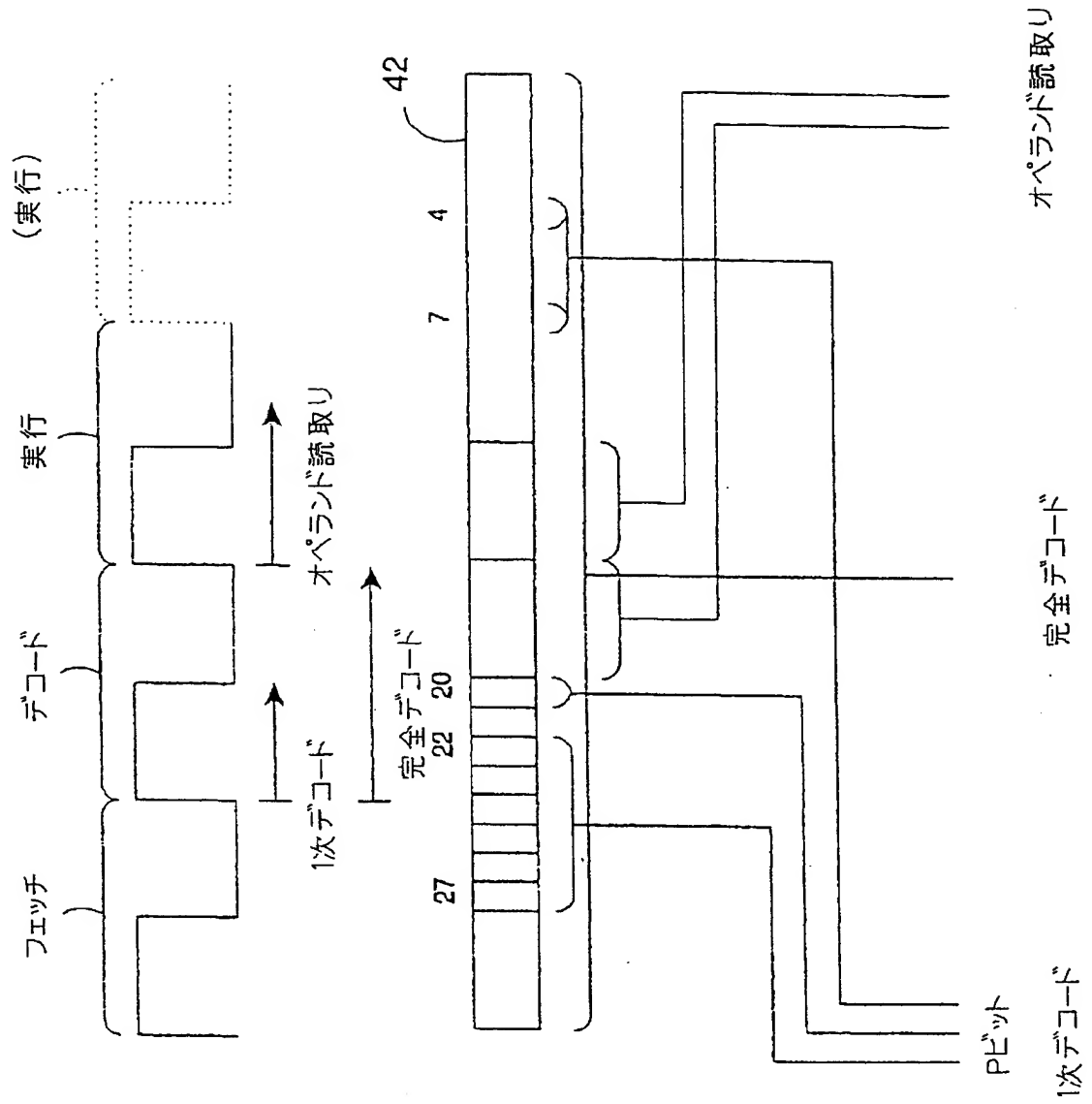
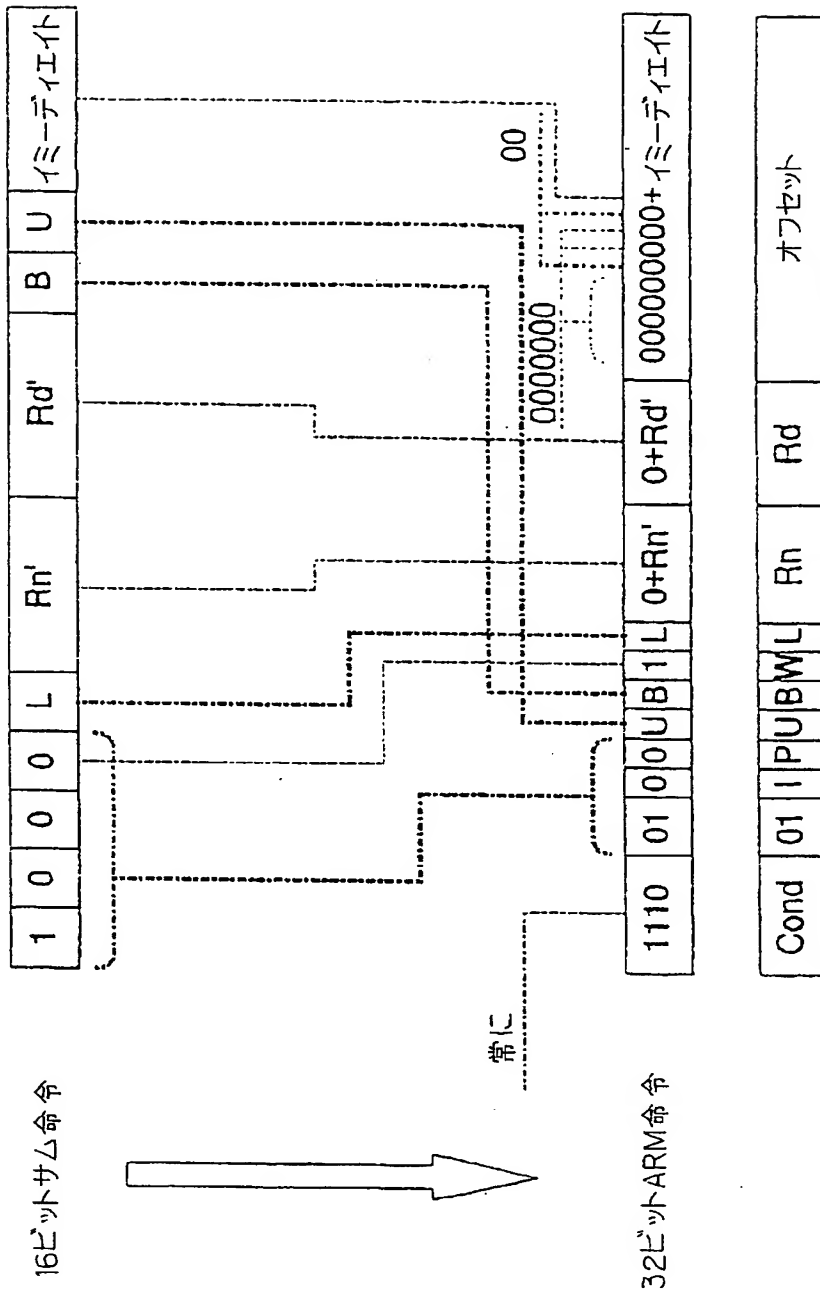


Fig.4



【 図 5 】

語(バイト)、符号付き3ビットイミディエイトオフセット、強制ライトバック、増/減後、をロード/記憶せよ



ARM単一データ転送フォーマット

Fig.5

【 図 6 】

語(バイト)、符号付きレジスタオフセット、ライトバックなし、増/減前、をロード/記憶せよ

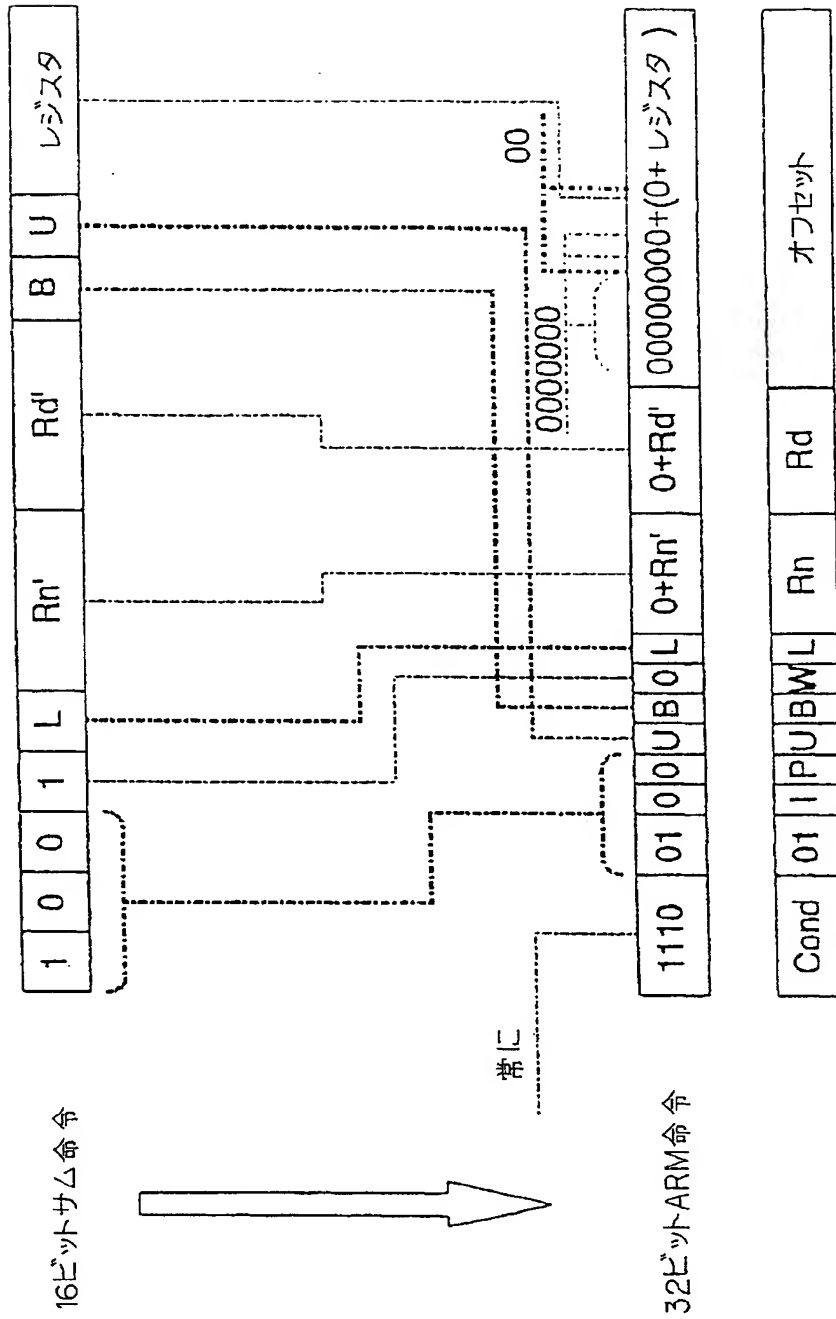


Fig.6

【 図 7 】

31	28	27	26	25	24	23	22	21	20	19	16	15	12	11	8	7	5	4	3	0		
Cond		00	1	演算コード				S	Rn			Rd			オペランド2							
Cond		000000				A		S	Rd			Rn			Rs		1001		Rm			
Cond		00010				B		00		Rn			Rd			0000		1001		Rm		
Cond		01	I	P	U	B	W	L	Rn			Rd			オフセット							
Cond		011	xxxxxxxxxxxxxxxxxxxx																	1	xxxx	
Cond		100	P	U	S	W	L	Rn			レジスタリスト											
Cond		101	オフセット																			
Cond		110	P	U	N	W	L	Rn			CRd		CP#		オフセット							
Cond		1110		CP Opc			CRn			CRd		CP#		CP		0		CRm				
Cond		1110		CP Opc			CRn			Rd		CP#		CP		1		CRm				
Cond		プロセッサにより無視される																				

Fig.7

【 図 8 】

15	14	13	12	11	10		8	7		5	4	3	2		0
0	0	0	00		Rd			Rs			0	OP	イミューディエイト		
0	0	0	00		Rd			Rs			1	OP	レジスタ		
0	0	0	OP=1-3		Rd			イミューディエイト							
0	0	0	OP=0-2		Rd			Rs			イミューディエイト				
0	0	1	1	1	OP1			Rd/Rs1			OP2		Rs2		
0	1	0	OP		Rd/Rs			イミューディエイト							
0	1	0	1	1	Rd			PC相対オフセット							
0	1	1	B	L	Rd/Rs			SP相対オフセット							
1	0	0	0	L	Rn			Rd/Rs			B	U	イミューディエイト		
1	0	0	1	L	Rn			Rd/Rs			B	U	レジスタ		
1	0	1	B	L	Rn			Rd/Rs			イミューディエイト				
1	1	0	0	L	Rn			レジスタリスト							
1	1	0	1	SP	Rd			有効アドレスオフセット							
1	1	1	0	Cond				8ビット分岐オフセット							
1	1	1	1	0	長い分岐およびリンクセグメント										
1	1	1	1	1	長い分岐およびリンクオフセット										

Fig. 8

【 図 9 】

サムレジスタ	ARMLレジスタ
一般レジスタ0	一般レジスタ0
一般レジスタ1	一般レジスタ1
一般レジスタ2	一般レジスタ2
一般レジスタ3	一般レジスタ3
一般レジスタ4	一般レジスタ4
一般レジスタ5	一般レジスタ5
一般レジスタ6	一般レジスタ6
一般レジスタ7	一般レジスタ7
	一般レジスタ8
	一般レジスタ9
スタックリミット	一般レジスタ10
	一般レジスタ11
	一般レジスタ12
スタックポインタ	スタックポインタ(R13)
リンクレジスタ	リンクレジスタ(R14)
プログラムカウンタ	プログラムカウンタ(R15)
CPSR	CPSR
SPSR	SPSR

Fig.9

## 【 国 際 調 査 報 告 】

## INTERNATIONAL SEARCH REPORT

A. CLASSIFICATION OF SUBJECT MATTER IPC 6 G06F9/318		Internat. Application No PCT/GB 95/00314
According to International Patent Classification (IPC) or to both national classification and IPC		
B. FIELDS SEARCHED		
Minimum documentation searched (classification system followed by classification symbols) IPC 6 G06F		
Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched		
Electronic data base consulted during the international search (name of data base and, where practical, search terms used)		
C. DOCUMENTS CONSIDERED TO BE RELEVANT		
Category *	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
Y	IBM TECHNICAL DISCLOSURE BULLETIN, vol. 15, no. 3, August 1972 NEW YORK US, page 920 J. C. KEMP 'Instruction Translator' see the whole document ---	1-4, 11-15
Y	IBM TECHNICAL DISCLOSURE BULLETIN, vol. 32, no. 10A, 1 March 1990 page 349 XP 000083346 'OPCODE REMAP AND COMPRESSION IN HARD-WIRED RISC MICROPROCESSOR' see the whole document ---	1-4, 11-15
A	EP, A, 0 109 567 (IBM) 30 May 1984 see the whole document ---	1, 15
-/--		
<input checked="" type="checkbox"/> Further documents are listed in the continuation of box C. <input checked="" type="checkbox"/> Patent family members are listed in annex.		
* Special categories of cited documents : "A" document defining the general state of the art which is not considered to be of particular relevance "E" earlier document but published on or after the international filing date "L" document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified) "O" document referring to an oral disclosure, use, exhibition or other means "P" document published prior to the international filing date but later than the priority date claimed "T" later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention "X" document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone "Y" document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art. "A" document member of the same patent family		
Date of the actual completion of the international search  31 May 1995		Date of mailing of the international search report  12.06.95
Name and mailing address of the ISA European Patent Office, P.O. 5818 Patentlaan 2 NL - 2280 HV Rijswijk Tel. (+ 31-70) 340-2040, Tx. 31 651 epo nl, Fax: (+ 31-70) 340-3016		Authorized officer  Daskalakis, T

## INTERNATIONAL SEARCH REPORT

Internal . Application No  
PCT/GB 95/00314

C.(Continuation) DOCUMENTS CONSIDERED TO BE RELEVANT		
Category	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
A	IBM TECHNICAL DISCLOSURE BULLETIN, vol. 17, no. 11, April 1975 NEW YORK US, pages 3438-3441, P. F. SMITH 'Extended Control for Microprocessors' see the whole document ---	1, 15
A	PATENT ABSTRACTS OF JAPAN vol. 007 no. 073 (P-186) ,25 March 1983 & JP,A,58 003040 (NIPPON DENKI KK) 8 January 1983, see abstract -----	

## INTERNATIONAL SEARCH REPORT

Information on patent family members

International Application No
PCT/GB 95/00314

Patent document cited in search report	Publication date	Patent family member(s)	Publication date
EP-A-0109567	30-05-84	WO-A- 8401635	26-04-84
		EP-A- 0124517	14-11-84
		US-A- 4587612	06-05-86
-----			